D^{2}F2WOD: Learning Object Proposals for Weakly-Supervised Object Detection via Progressive Domain Adaptation

Yuting Wang
Rutgers University
Piscataway, NJ
yw632@cs.rutgers.edu

Ricardo Guerrero
Samsung AI Center
Cambridge, UK
r.guerrero@samsung.com

Vladimir Pavlovic
Rutgers University
Piscataway, NJ
vladimir@cs.rutgers.edu

Abstract

*Weakly-supervised object detection (WSOD) models attempt to leverage image-level annotations in lieu of accurate but costly-to-obtain object localization labels. This oftentimes leads to substandard object detection and localization at inference time. To tackle this issue, we propose D^{2}F2WOD, a **D**ual-**D**omain **F**ully-to-**W**eakly Supervised **O**bject **D**etection framework that leverages synthetic data, annotated with precise object localization, to supplement a natural image target domain, where only image-level labels are available. In its warm-up domain adaptation stage, the model learns a fully-supervised object detector (FSOD) to improve the precision of the object proposals in the target domain, and at the same time learns target-domain-specific and detection-aware proposal features. In its main WSOD stage, a WSOD model is specifically tuned to the target domain. The feature extractor and the object proposal generator of the WSOD model are built upon the fine-tuned FSOD model. We test D^{2}F2WOD on five dual-domain image benchmarks. The results show that our method results in consistently improved object detection and localization compared with state-of-the-art methods.*

1. Introduction

Object detection has achieved remarkable progress over the past few years, mostly through the development of deep neural network architectures [20, 4]. However, training such deep neural networks needs large amounts of manually annotated images. Obtaining these annotations is costly and time-consuming. Thus, reducing these costs is of great importance, and many weakly-supervised object detection (WSOD) methods [2, 29, 28] have been developed accordingly. WSOD methods alleviate the reliance on precise object localization information by training detection architectures using only *image-level* annotations.

Most existing WSOD algorithms [2, 29, 28, 34, 21, 8, 13] are based on multiple instance learning (MIL) [6]. They treat images as bags of object proposals, which are produced by an object proposal generator [31, 41]. Although many promising results have been achieved by WSOD, they are still not comparable to fully-supervised object detectors (FSOD) [20, 4]. One of the main reasons is that state-of-the-art object proposal generators still cannot produce accurate object proposals – this is a particularly serious issue for *in-the-wild* images with multiple complex non-rigid objects and cluttered background, as shown in Fig. 1.

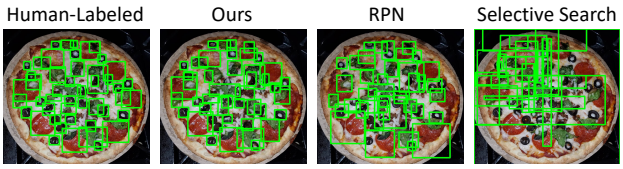

Figure 1: Illustration of human-labeled objects contrasted to object proposals generated by our D^{2}F2WOD$_{\text{warm-up}}$, RPN of Faster R-CNN trained on synthetic data alone, and Selective Search (SS), on the RealPizza10 dataset. It demonstrates the benefit of our learned object proposal generator (warm-up stage) over SS. SS often fails to generate accurate bounding boxes, making it hard to improve classification accuracy. It also shows that our D^{2}F2WOD$_{\text{warm-up}}$ is better than learned RPN of Faster R-CNN trained on synthetic data alone. Our warm-up domain adaptation stage can improve the precision of the object proposals in the target domain.

To overcome this difficulty, we introduce a simple object proposal generation strategy that can be applied to different WSODs to improve their detection performance. Our key insight is to cast WSOD as a *domain adaptation problem* – while target "natural" images often lack localization labels, localization is "freely" available for "non-photographic" synthetic images. For instance, when synthesizing images such as SyntheticPizza10 [19], localization and identity labels are available as a byproduct of the generation process. Highly stylized images (*e.g.*, Clipart1K [14], Watercolor2K [14], and Comic2K [14]) are likewise easier to annotate than natural images, where objects may exhibit com-

plex changes in share or appearance. In this work, we are interested in leveraging fully-annotated non-photographic datasets to support accurate object localization in real-world datasets. To this end, we propose a **D**ual-**D**omain **F**ully-to-**W**eakly Supervised **O**bject **D**etection (D^2F2WOD) framework, which is able to produce accurate object proposals using image-level labels of natural images along with fully-supervised non-photographic images through *progressive domain adaptation* of an FSOD model.

Given the large domain gap between the source and the target, across both foreground and background (F&B), it is critical to (1) individually address the adaptation of F&B in a disentangled manner when feasible, and (2) reduce the domain gap in a gradual manner to control the propagation of errors. In our work, we progressively adapt an FSOD model from source images to the target domain in five steps. First, we build an initial bridge between the non-photographic source and the real-world target domains using unpaired image-to-image translation (I2I), such as [40]. This creates "target-like" intermediate images with location-accurate object instances but divergent appearance. Instead of the common practice of initializing an FSOD on this intermediate domain, we further reduce the domain gap by employing a copy-paste augmentation technique sourced in [35, 9] to fuse the translated object appearance with real background images and create a second transfer-labeled intermediate domain. This domain serves as the preliminary stage for initializing an FSOD, to be used for pseudo labeling (PL) [16] in the subsequent WSOD learning phase on the real target domain. However, the typical number of confident pseudo-labeled instances resulting from the initialized FSOD and needed for WSOD is insufficient for effective adaptation to the target domain. To that end, we re-employ the previously used augmentation technique to increase the number of confident PL instances. Finally, we learn a detection head utilizing these target-like object proposal features. Our D^2F2WOD achieves consistent improvements compared with the state-of-the-art methods, offering a strong baseline for WSOD models.

Our contributions are three-fold: (1) We propose a framework for object proposal generation based on domain adaptation, applicable to different WSODs, including OICR [29], and CASD [13]. The five-step progressive domain adaptation process exploits gradual adaptation of the FSOD on generated samples, as well as with decoupled focus on foreground and background, and it can be seamlessly integrated with different types of FSOD backbones such as Faster R-CNN [20] and transformer-based detectors [4]. (2) We construct a dual-domain image benchmark SyntheticPizza10 $\rightarrow$ RealPizza10 with non-photographic images as the source and real-world images as the target domains. (3) The experimental results show that our D^2F2WOD achieves state-of-the-art performance on five benchmarks.

2. Related Work

Weakly-Supervised Object Detection. WSOD methods generally aim to exploit only image-level annotations, as opposed to the fine-grained object localization usually used in FSOD. Existing methods mainly cast WSOD as a multiple-instance learning (MIL) problem, where objects are not necessarily centered in images and there is cluttered background [18]. In MIL-based models, an image is interpreted as a bag of potential object instances. These models generally consist of three components: feature extractor (FE), object proposal (OP) generator, and detection head (DH). Given an image, they first feed it into the OP generator and the FE to generate proposals and features maps, respectively. Then, the feature maps and object proposals (OPs) are forwarded into a Spatial Pyramid Pooling (SPP) layer [32] or a Region-of-Interest (RoI) pooling layer [20] to produce fixed-size object proposal features. Finally, these feature vectors are fed into the DH to classify and localize objects. End-to-end weakly-supervised deep detection network (WSDDN) [2] proposes one of the first MIL frameworks. Based on Fast R-CNN [10], it introduces a two-stream network to perform classification and localization, respectively. However, in WSDNN, the top ranking OPs may only cover the most discriminative parts of the objects instead of whole object instances, due to a lack of supervision in terms of precise localization information in the training process. Subsequent work [29, 28, 34, 1, 36, 21, 8, 13, 30] aims to alleviate this problem by extending WSDDN. One of the key factors that affect the performance of WSOD is the quality of OPs. Many existing methods are built upon unsupervised RoI extraction, such as selective search (SS) [31] and edge boxes (EB) [41]. To generate OPs, SS uses both exhaustive search and segmentation, and EB uses object edges. [37] proposes a hierarchical region proposal refinement network and [30] proposes a two-stage region proposal network, to refine proposals gradually. Some other work, such as W2N [12], continues refine the noisy dataset generated by a well-trained WSOD with semi-supervised learning.

Different from the above methods, in this work, we first cast WSOD as a domain adaptation problem, by leveraging an auxiliary source domain to pre-train an FSOD model. The FSOD model is progressively adapted from the source to the target domains. After we obtain the adapted FSOD model, we treat it as the weakly-supervised OP generator in the WSOD settings, and at the same time the FE of the FSOD is treated as the pre-trained FE for the WSOD model.

Domain Adaptation for Object Detection. Domain adaptation typically involves two domains, namely source and target domains. Most of existing domain adaptation methods aim to address the domain shift between a fully-labeled source domain and an unlabeled or weakly-labeled target domain, which is formulated as unsupervised or weakly-

supervised domain adaptation, respectively. State-of-the-art domain adaptation for object detection introduces different strategies to reduce the domain divergence. For example, adversarial feature learning is leveraged to adapt object detectors to a target domain with the help of a domain discriminator [5, 25, 26, 11, 33], thus producing domain invariant features. Highly confident predictions generated by a source detector are used as pseudo-labels to fine-tune the detector on the target domain [14, 15, 39, 24]. Similarly, an unpaired I2I model [14, 22, 11] can be employed to map a source image to a target-like image. Introducing this target-like domain mitigates the difficulty of direct transfer between source and target with a large domain gap.

Different from the aforementioned approaches, our method decouples the domain shift into the foreground and background shift. This makes it possible to gradually, in a focused manner, adapt the detector from source to target. We also use data augmentation in the adaptation stage, since augmentations such as color jittering [27], mixup [38] and copy-paste [35, 9] can have major impact on image classification and object detection. Furthermore, OPs generated by the adapted object detector are augmented by an additional refinement of proposal branches using the detection heads in the WSOD settings. This refinement improves the network's ability to classify and localize the OPs.

3. Methodology

The proposed Dual-Domain Fully-to-Weakly Supervised Object Detection framework (D^{2}F2WOD) aims to address the lack of object localization information in the target domain by formulating WSOD as a domain adaptation problem. It decouples WSOD model training into two stages – domain adaptation and WSOD. In the domain adaptation stage, we *progressively* learn a domain-adaptive FSOD by leveraging an auxiliary source domain as warm-up. In the WSOD stage, this adapted FSOD is used to initialize the WSOD model, which is then refined on the target domain. D^{2}F2WOD is a general framework that can employ different FSOD and WSOD methods. Here, we focus on two representative FSOD backbones – Faster R-CNN [20] and DETR [4], and two representative WSOD models – the widely-used OICR [29] and the state-of-the-art CASD [13]. In this section, we first formulate the problem, followed by the framework overview, the details of the architecture, and the training procedure for each stage of D^{2}F2WOD.

3.1. Problem Formulation

Fig. 2 illustrates our D^{2}F2WOD approach. Our goal is to detect object instances in a real-world, weakly-supervised target domain $\mathcal{T}$ (*e.g.*, real pizza in Fig. 2) by leveraging a non-photographic source domain $\mathcal{S}$ (*e.g.*, synthetic pizza in Fig. 2). For this problem, we have access to images with only image-level annotations (*i.e.*, class labels) in $\mathcal{T}$ and images with rich instance-level annotations (*i.e.*, class labels and bounding boxes) in $\mathcal{S}$.

Formally, $\mathbf{X}_s \in \mathbb{R}^{h\times w\times 3}$ denotes an RGB image from $\mathcal{S}$, where h and w are the height and width of the image, respectively. $\mathbf{Y}_s^{(f)} = \{(\mathbf{b}_1, c_1), \dots, (\mathbf{b}_{N_s}, c_{N_s})\}$ indicates the instance-level full-annotation associated with $\mathbf{X}_s$, where $\mathbf{b}_i \in \mathbb{R}^4$ is the i-th object localization bounding box defined by $[x_{\min}, y_{\min}, x_{\max}, y_{\max}]$ that specifies its top-left corner $(x_{\min}, y_{\min})$ and its bottom-right corner $(x_{\max}, y_{\max})$, and $c_i \in \{1, \dots, C\}$ is its category label. N_s is the number of object instances associated with $\mathbf{X}_s$. The classes to be detected in $\mathcal{T}$ are shared with $\mathcal{S}$, and C is the number of object categories in the two domains. Similarly, $\mathbf{X}_t \in \mathbb{R}^{h\times w\times 3}$ denotes an RGB image from $\mathcal{T}$, and $\mathbf{Y}_t^{(w)} = [y_1, \dots, y_C] \in \{0, 1\}^C$ denotes the image-level weak-supervision, where $y_c \in \{0, 1\}$ indicates the absence (presence) of at least one instance of c-th category. N_t is the number of present object classes associated with $\mathbf{X}_t$. We denote $\mathbf{V}_j$ as object proposal feature vectors of images from domain $j \in \{\mathcal{S}, \mathcal{T}\}$. In this work, we aim to learn an object detector for the target domain, $\hat{\mathbf{Y}}^{(f)} = f(\mathbf{X}|\boldsymbol{\theta}), \mathbf{X} \in \mathcal{T}$, by leveraging both the fully annotated data $\mathcal{D}_s = \{(\mathbf{Y}_s^{(f)}, \mathbf{X}_s)\}$ from $\mathcal{S}$ and the weakly annotated data $\mathcal{D}_t = \{(\mathbf{Y}_t^{(w)}, \mathbf{X}_t)\}$ from $\mathcal{T}$; in other words, $\boldsymbol{\theta}^* \leftarrow \mathcal{D} = \{\mathcal{D}_s \cup \mathcal{D}_t\}$.

3.2. Approach Overview

To boost the performance of a WSOD model on $\mathcal{T}$, our key insight is to jointly improve the precision of the OPs, and learn *target-domain-specific* and *detection-aware* proposal features. To this end, our D^{2}F2WOD exploits the fully-labeled $\mathcal{S}$ (FLS) domain and introduces a dual-stage training scheme as shown in Fig. 2. In the *warm-up* domain adaptation stage, an FSOD is pre-trained (PT) on $\mathcal{S}$ and progressively fine-tuned (FT) on (1) a transfer-labeled intermediate (TLI) domain $\mathcal{G}_1$, (2) an augmented (Aug.) transfer-labeled intermediate domain $\mathcal{G}_2$, and then on (3) the pseudo-labeled target (PLT) domain $\mathcal{T}$ and (4) the augmented pseudo-labeled target domain $\mathcal{T}$. As shown in Fig. 2, $\mathcal{G}_1$ is constructed as target-like instances with accurate transferred localization information, and $\mathcal{G}_2$ is constructed as target-like images with accurate transferred localization information and real background, thus bridging the $\mathcal{S}$ and $\mathcal{T}$ and facilitating the adaptation. In the *main* WSOD stage, an MIL-based WSOD model is specifically tuned to $\mathcal{T}$. The FE and the OP generator of the WSOD model are built upon the fine-tuned FSOD model.

3.3. Warm-Up Domain Adaptation Stage: Learning Domain-Specific Features & Object Proposals

The warm-up stage of D^{2}F2WOD trains an FSOD model on the dual $\mathcal{S}$-$\mathcal{T}$, which provides the pre-trained deep FE and OP generator to produce object proposal feature vectors on $\mathcal{T}$. This FSOD model is later used in the main

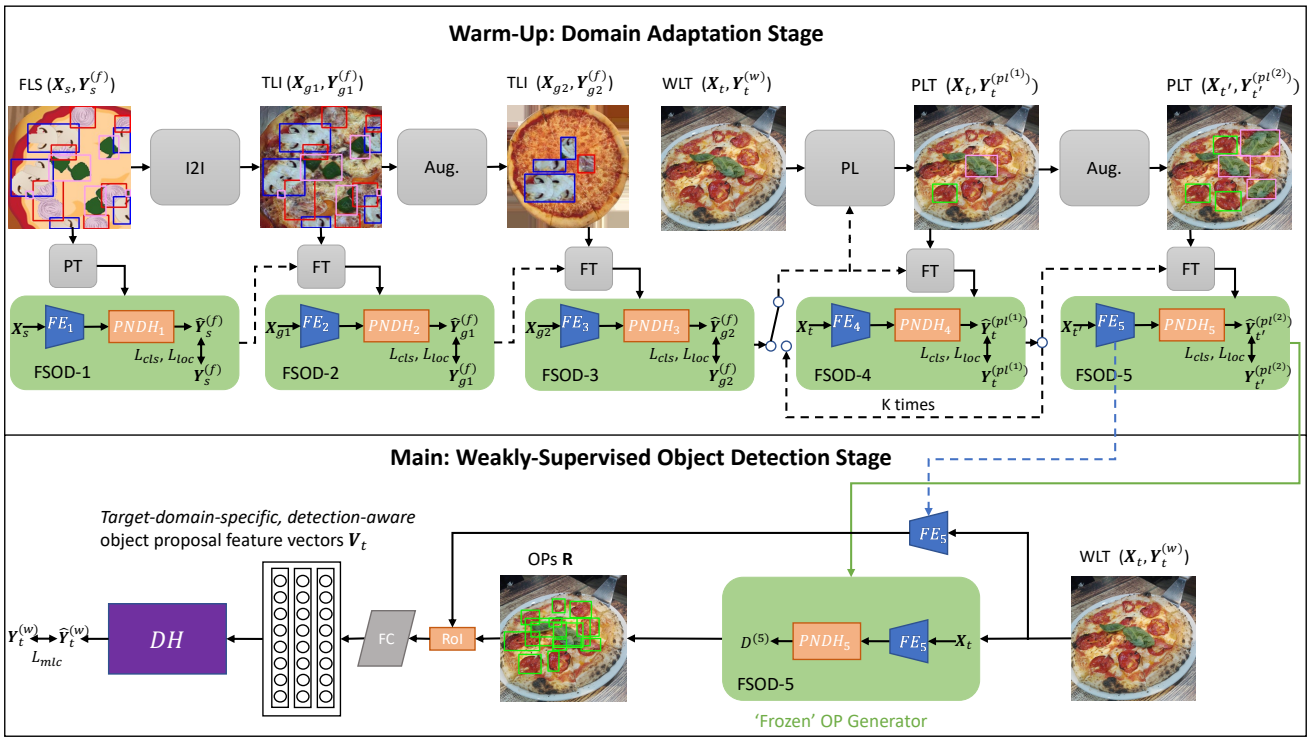

Figure 2: An overview of our Dual-Domain Fully-to-Weakly Supervised Object Detection Architecture (D^{2}F2WOD). **Upper block:** warm-up domain adaptation stage; **lower block:** main weakly-supervised object detection stage. Here 'A- - - B' denotes that the parameters of module B are initialized from module A's parameters. 'A——B' denotes that the output of module A is used as an input of module B or module B directly copies module A's parameters without further fine-tuning. 'PNDH' denotes the proposal networks and detection heads in an FSOD model.

stage for initializing the WSOD model. Our method generalizes across different FSODs. Here, we adopt two architectures for the FSOD model – Faster R-CNN [20] and Sparse DETR [23] from the DETR [4] family of object detectors[1].

3.3.1 Progressive Domain Adaptation.

Directly training the FSOD model on the dual domain is challenging, because of (1) substantial data distribution shift between source (non-photographic images) and target (natural images) domains in both foreground and background, and (2) significant supervision discrepancy between source (fully-labeled) and target (lack of localization information) domains. Inspired by DT+PL [14], we overcome this difficulty by generating an intermediate domain $\mathcal{G}_1$ and $\mathcal{G}_2$ with instance-level annotations transferred from the source domain. Correspondingly, we introduce a five-step progressive domain adaptation strategy (upper part in Fig. 2) that first pre-trains the FSOD-1 model on the fully-labeled $\mathcal{S}$, and gradually fine-tunes it on the transfer-labeled $\mathcal{G}_1$ and augmented transfer-labeled $\mathcal{G}_2$ to be FSOD-2 and FSOD-3, and then on the first-round pseudo-labeled $\mathcal{T}$ and second-round augmented pseudo-labeled $\mathcal{T}$ to be FSOD-4 and FSOD-5.

Automated generation of intermediate domains for initial adaptation. To facilitate the adaptation, a desired property of the intermediate domain should be that *its images are similar to the target images while having accurate localization information*. To this end, we generate the intermediate domain images as composition of photo-realistic, target-like objects guided by the layout of objects in the source images, thus allowing direct transfer of localization annotations from the source images to the generated images.

Specifically, since there are no corresponding image pairs between $\mathcal{S}$ and $\mathcal{T}$ domains, we train CycleGAN [40], an unpaired I2I network, to map source images $\mathbf{X}_s$ to domain $\mathcal{G}_1$ intermediate to the target $\mathcal{T}$:

$$\mathbf{X}_{g_1} = f_{\mathcal{S}\to\mathcal{G}_1}(\mathbf{X}_s), \tag{1}$$

where $\mathbf{X}_{g_1} \in \mathbb{R}^{h\times w\times 3}$ is the image generated by I2I network. Given this I2I mapping, we transfer the labels from instances in $\mathbf{X}_s$ to those in $\mathbf{X}_{g_1}$ as

$$\mathbf{Y}_{g_1}^{(f)} = \mathbf{Y}_s^{(f)} : \quad \mathbf{X}_{g_1} = f_{\mathcal{S}\to\mathcal{G}_1}(\mathbf{X}_s). \tag{2}$$

[1] Sparse DETR enhanced the efficiency of DETR and improved the performance on small objects datasets, and thus we choose Sparse DETR here.

Using the intermediate images $\mathbf{X}_{g_1}$ together with their instance-level annotations $\mathbf{Y}_{g_1}^{(f)}$, we fine-tune the FSOD-1 model, pre-trained on $\mathcal{S}$, into FSOD-2.

To make $\mathbf{X}_{g_1}$ closer to the $\mathbf{X}_t$ images, we focus on separately bridging the foreground and background gap. Specifically, we employ an object-aware data augmentation based on copy-paste [35] to map $\mathbf{X}_{g_1}$ to $\mathbf{X}_{g_2}$ images. For each $\mathbf{X}_{g_1}$ image, we randomly copy several foreground object instances from $\mathcal{G}_1$, with resizing and flipping transformations, and paste them onto the real-world target background images from $\mathcal{T}$ to generate $\mathbf{X}_{g_2}$. Using the augmented intermediate images $\mathbf{X}_{g_2}$ together with their instance-level annotations $\mathbf{Y}_{g_2}^{(f)}$, we fine-tune the FSOD-2 model to FSOD-3.

Instance-level pseudo-annotation of target images for continual adaptation. While the intermediate domains $\mathcal{G}_1$ and $\mathcal{G}_2$ partly bridge the source and target domains, there is still non-negligible domain shift between the intermediate and target domains. For example, the synthesized objects translated via CycleGAN are still different from those in the target images; the layout of objects in the intermediate domain is restrictive to that in $\mathcal{S}$ and lacks the real-world variation in $\mathcal{T}$. Therefore, to achieve good detection performance on the target domain, we need to further fine-tune the FSOD-3 model on the target domain $\mathcal{T}$ as FSOD-4. For this purpose, we use FSOD-3, initially fine-tuned on $\mathcal{G}_2$, to produce instance-level pseudo-annotations $\mathbf{Y}_t^{(pl^{(1)})} = \{(\mathbf{b}_1, c_1), \ldots, (\mathbf{b}_{N_t}, c_{N_t})\}$ for each weakly-labeled target (WLT) image $\mathbf{X}_t$.

Specifically, for each image $\mathbf{X}_t$, we first obtain the predictions $D^{(3)}$ from the FSOD-3 model:

$$D^{(3)} = \{D_1, ..., D_C\} = f_{\text{FSOD-3}}(\mathbf{X}_t), \tag{3}$$

where D_j indicates all predictions belonging to class $j \in \{1, \ldots, C\}$. $D_j = \{d_1, ..., d_{N_j}\}$, N_j is the number of class j detections, $d_m = (p_m, \hat{\mathbf{b}}_m, j)$, and $p_m \in \mathbb{R}$ indicates the probability of detection $\hat{\mathbf{b}}_m$ belonging to class j. For each ground-truth object class c, we select the top-1 confident prediction d_q from D_c, and we add $(\hat{\mathbf{b}}_q, c)$ to $\mathbf{Y}_t^{(pl^{(1)})}$:

$$d_q = (p_q, \hat{\mathbf{b}}_q, c): \quad y_c = 1, \quad q = \underset{m}{\operatorname{argmax}}\, p_m. \tag{4}$$

The FSOD-3 model is subsequently fine-tuned on the target images $\mathbf{X}_t$ with instance-level pseudo-annotations $\mathbf{Y}_t^{(pl^{(1)})}$ into FSOD-4, finally adapting from the target-like $\mathcal{G}_2$ to $\mathcal{T}$. In principle, it can be performed K times to generate instance-level pseudo-annotations $\mathbf{Y}_t^{(pl^{(k)})}$ and adapted into FSOD-$(3+k)$, where $k \in \{1, \ldots, K\}$.

However, the typical number of confident pseudo-labeled instances resulting from the FSOD-3 is insufficient for effective adaptation to the target domain. To add instances annotations, we use the copy-paste augmentations again to produce those object instances. We repeat the previous step, in which the FSOD-4 model is used to produce instance-level pseudo-annotations $\mathbf{Y}_t^{(pl^{(2)})}$. For each pseudo-labeled instance $(\hat{\mathbf{b}}_q, c)$ in $\mathbf{X}_t$, we copy and randomly paste it L times as $\{(\hat{\mathbf{b}}_{q_1}, c)..., (\hat{\mathbf{b}}_{q_L}, c)\}$ onto the original target image $\mathbf{X}_t$ and produce an augmented image $\mathbf{X}_t^{'}$ with new pseudo-annotations $\mathbf{Y}_t^{(pl^{(2)})}$. The FSOD-4 is subsequently fine-tuned on the augmented targets $\mathbf{X}_t^{'}$ with instance-level pseudo-annotations $\mathbf{Y}_t^{(pl^{(2)})}$ into FSOD-5, thus adapting from $\mathcal{T}$ to the augmented $\mathcal{T}$.

3.4. Main WSOD Stage: Classification and Localization Refinement of Object Proposals

In the main stage of D^{2}F2WOD, we exploit the FSOD-5 model obtained in the warm-up stage to initialize a WSOD model and train it on the real-world target data. As explained in Sec. 2, an MIL-based WSOD model consists of a FE, an OP generator, and a DH. We initialize the FE of the WSOD model with the FE of the fine-tuned FSOD-5 (blue block in Fig. 2) and continually train it on $\mathcal{T}$. We replace the standard selective search based OP generator of the WSOD model by the entire fine-tuned FSOD-5 (green block in Fig. 2), which is not trained in the WSOD training procedure. Note that here we treat the detection output of the FSOD-5 as the object proposals of the WSOD. This strategy can be seamlessly applied to different types of WSODs, and here we consider the widely-used OICR [29] and the state-of-the-art CASD [13]. By doing so, our proposed model significantly outperforms existing WSOD methods due to: (1) target-domain-specific pre-trained features, (2) detection-aware pre-trained features, and (3) target-domain-specific object proposals.

Generating object proposals and its features. Given an image $\mathbf{X}_t$, the OP generator aims to obtain M_t bounding boxes $\mathbf{R} = \{\mathbf{b}_1, ..., \mathbf{b}_{M_t}\}$ associated with $\mathbf{X}_t$. To this end, for each image $\mathbf{X}_t$, we first obtain the predictions $D^{(5)}$ from the fine-tuned FSOD-5 model:

$$D^{(5)} = \{D_1, ..., D_C\} = f_{\text{FSOD-5}}(\mathbf{X}_t). \tag{5}$$

Given that the number of predictions produced by DETR is much less than that of Faster R-CNN, we adopt different proposal generation strategies. For DETR, all predicted bounding boxes are added to $\mathbf{R}$. For Faster R-CNN, we select predicted bounding boxes $\hat{\mathbf{b}}_m$ belonging to the ground-truth classes to $\mathbf{R}$. Using the FE followed by an RoI pooling layer and two fully-connected (FC) layers for the WSOD model, we then obtain d-dimensional object proposal feature vectors $\mathbf{V}_t \in \mathbb{R}^{d \times M_t}$ for each input image $\mathbf{X}_t$ (lower part in Fig. 2).

Classification and localization refinement of object proposals. These object proposal feature vectors $\mathbf{V}_t$ are fed

into the detection head of OICR [29] or CASD [13] to classify and localize objects. Please refer to the Sec. 1 of supplementary material for the details.

4. Experimental Results

Benchmarks. We evaluate our method on five dual-domain image benchmark pairs: SyntheticPizza10 [19] → RealPizza10 [19], Clipart1K [14] → VOC2007 [7], Watercolor2K [14] → VOC2007-sub, Comic2K [14] → VOC2007-sub, and Clipart1K → MS-COCO-sub [17] datasets. We construct the SyntheticPizza10 dataset from [19] by including single-layer images and removing the pizza base-only images (*i.e.*, without any toppings). RealPizza10 is a subset of the PizzaGAN [19], containing 9,213 real images annotated with 13 toppings. As we use pseudo-labeling, we require the classes in $(\mathcal{S}, \mathcal{T})$ to be the same. Thus, we remove images from the PizzaGAN dataset having only spinach, arugula, or corn, the classes absent from SyntheticPizza10, to construct RealPizza10. Similarly, MS-COCO-sub and VOC2007-sub datasets are constructed by removing images without having at least one class from the $\mathcal{S}$ domains. Please see the Sec. 2 of the supplementary material for details.

The number of instances for each class in each pair of datasets is unbalanced. Compared with the other benchmarks, SyntheticPizza10 → RealPizza10 are more challenging since all Pizza object instances are quite small and have diverse shape and texture. Although [19] uses a variety of different clip-art images for each topping to obtain the synthetic pizzas as shown in Fig. 3, the number of these ingredient templates is still limited. In a real food image, the shape, color and texture of each ingredient object are dependent on cooking actions. As shown in Fig. 3, for each ingredient, the domain gap between SyntheticPizza10 and RealPizza10 varies. In addition, the gap extends to bases of synthetic and real pizzas, as shown in Fig. 3.

Baselines and Evaluation Procedure. We mainly focus on comparing against the state-of-the-art DAOD baselines (cross-domain): **DT+PL** [14] and **PADOD** [11], and widely-used WSOD baselines (single-domain): **OICR** [29], **CASD** [13], and other baselines including WSDDN [2], PCL [28], C-MIL [34], WSOD2(+Reg) [36], Pred Net [1], C-MIDN [8], MIST(+Reg) [21], WeakRPN [30], CASD^2 (training CASD two times: once for proposal and once for object detection), and CASD+W2N [12]. Our evaluation follows the standard detection procedure. We compute Average Precision (**AP**) and the mean of AP (**mAP**) as the evaluation metric. A predicted box is treated as a positive example if it has an IOU > 0.5 between ground truth bounding boxes and the predicted box.

Implementation Details. In the warm-up stage, Faster R-CNN [20] and Sparse DETR [23] were used as our FSOD models. For each target image, we generated 438 object proposals per image on average on VOC2007 and 441 object proposals per image on average on RealPizza10. Please refer to the Sec. 3 in the supplementary material for details.

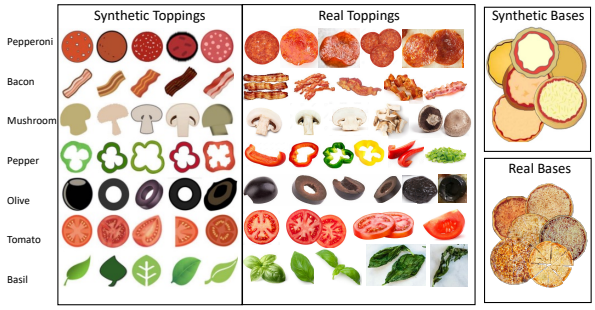

Figure 3: There is a domain shift in toppings and bases of pizzas. Left: Examples of toppings used to create synthetic pizza images [19]. Middle: Examples of toppings in real pizza images. Right top: Examples of bases used to create synthetic pizza images [19]. Right bottom: Examples of bases in real pizza images.

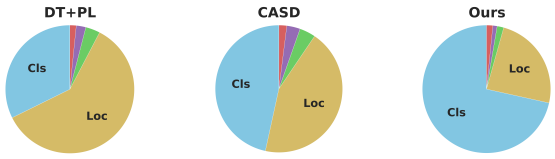

Figure 4: Identify object detection errors.

Source vs. Target Labeling Cost. Two factors determine the trade-off of source vs. target FSOD: the cost of building a synthetic image generator and the realism of the synthesized images. When the realism of the synthetic images is moderate, the cost of building the generator is low. For SyntheticPizza10, built from abstract clipart or patches, the cost of generation is low. Moreover, the annotation of synthetic images is either a byproduct of the generation or inherently easy for human annotators, if such annotation is needed (*e.g.*, Clipart1k). Thus, our approach has inherently lower cost than the direct annotating the target domain.

4.1. Main Results

We compare $\text{D}^2\text{F2WOD}$ with state-of-the-art single (SD) and cross-domain (CD) methods in terms of mAP. Table 1 and Table 2 summarize the detection results on five benchmarks based on Faster R-CNN FSOD backbone. The per class APs are listed in the supplementary material Table 5. $\text{D}^2\text{F2WOD}$ incorporated with OICR is denoted as $\text{D}^2\text{F2WOD}_{\text{oicr}}$, and with CASD is denoted as $\text{D}^2\text{F2WOD}_{\text{casd}}$. The results of our warm-up stage are denoted as $\text{D}^2\text{F2WOD}_{\text{warm-up}}$.

D^2F2WOD consistently outperforms the SD baselines. As shown in Table 1, on Clipart1K → VOC2007, $\text{D}^2\text{F2WOD}_{\text{casd}}$ reaches 64.8% mAP, outperforming the original CASD by 7.8% mAP, and $\text{D}^2\text{F2WOD}_{\text{casd+w2n}}$ reaches 66.9% mAP, out-

Table 1: Results (mAP in %) for different methods on Clipart1K → VOC2007. We denote as Upper-Bound the FSOD (Faster R-CNN or Sparse DETR) results, trained and tested on *fully-annotated* target domain to indicate the weak upper-bound performance of our methods. Our warm-up stage is compared with CD models and our main stage is compared with SD models. Faster R-CNN in CD means that we trained our network on fully-annotated source and test on fully-annotated target domains. The best and second best results for D^{2}F2WOD compared with baselines are shown in red and blue.

		After Warm-Up Stage				After Main Stage												
Type		CD			Ours	SD											Ours	
Method	Upper-Bound	Faster R-CNN [20]	DT+PL [14]	PADOD [11]	D^{2}F2WOD$_{\text{warm-up}}$	WSDDN [2]	OICR [29]	PCL [28]	WeakRPN [30]	C-MIL [34]	WSOD2(+Reg) [36]	Pred Net [1]	C-MIDN [8]	MIST(+Reg) [21]	CASD [13]	CASD*	CASD+W2N [12]	D^{2}F2WOD$_{\text{casd}}$	D^{2}F2WOD$_{\text{casd+w2n}}$
mAP	69.9	22.8	34.6	24.2	37.3	34.8	41.2	43.5	45.3	50.5	53.6	52.9	52.6	54.9	57.0	57.4	65.4	64.8	66.9

Table 2: Results (mAP in %) for different methods on SyntheticPizza10 → RealPizza10 (SPizza → RPizza), Watercolor2K → VOC2007-sub (Water → VocS), Comic2K → VOC2007-sub (Comi → VocS), and Clipart1K → MS-COCO-sub (Clip → CocoS).

			After Warm-Up Stage				After Main Stage		
	Type		CD			Ours	SD		Ours
	Method	Upper-Bound	Faster R-CNN [20]	DT+PL [14]	PADOD [11]	D^{2}F2WOD$_{\text{warm-up}}$	OICR [29]	CASD [13]	D^{2}F2WOD$_{\text{casd}}$
mAP	SPizza → RPizza	-	4.3	14.9	8.1	17.9	4.7	12.9	25.1
	Water → VocS	*78.0*	42.1	49.4	-	52.1	-	65.2	73.2
	Comi → VocS	*78.0*	33.5	46.5	-	49.6	-	65.2	70.8
	Clip → CocoS	*84.3*	13.9	22.1	-	25.7	-	48.3	57.2

performing the original CASD+W2N by 1.5% mAP, while CASD[2] outperforms the original CASD only by 0.4% mAP. The detection performance does not benefit much from using CASD[2], since doing so does not improve the generated proposals. On SyntheticPizza10 → RealPizza10 reported in Table 2, D^{2}F2WOD$_{\text{casd}}$ provides a 12.2% improvement over the original CASD in terms of mAP. **D^{2}F2WOD also consistently outperforms the CD baselines.** Table 1 shows that on Clipart1K → VOC2007 D^{2}F2WOD$_{\text{casd}}$ outperforms DT+PL and PADOD by 30.2% and 40.6% mAP, respectively. As shown in Table 2, D^{2}F2WOD$_{\text{casd}}$ outperforms DT+PL and PADOD on SyntheticPizza10 → RealPizza10 by 10.2% and 17.0% mAP, respectively. **D^{2}F2WOD generalizes across different datasets.** As shown in Table 2, D^{2}F2WOD effectively handles different domain shifts, successfully leveraging a variety of $\mathcal{S}^2$.

We observe both stages of D^{2}F2WOD yield consistently improved detection and localization performance compared with both state-of-the-art SD and CD baselines, especially on the more challenging SyntheticPizza10 → RealPizza10 scenario. By exploiting our domain adaptation stage, we believe that our training of the WSOD model is superior to existing methods in three important ways. First, our pre-trained features are target-domain-specific, because of progressive adaptation from source to intermediate to target domains, whereas existing WSOD methods use features pre-trained on ImageNet. Second, our pre-trained features are detection-aware, while ImageNet features used in existing WSOD methods are pre-trained with a single whole-image classification loss, which encourages translation and scale-invariant features. In contrast, the training of our FSOD model involves classification and regression losses, providing features that are sensitive to object locations and scales and are thus useful for detection. Third, our object proposals are target-domain-specific and of high-quality, since they are progressively learned directly on the target domain from foreground and background. Existing WSOD methods use hand-crafted selective search object proposals, which leads to inaccurate proposals especially for domains such as Pizza, with properties different from VOC2007.

[2]Resource constraints limit our focus on best select SOTA, with extensive comparison delegated to Clipart1K → VOC2007 evaluation.

4.2. Ablation Study

We first conducted ablation studies to investigate the effectiveness of our warm-up stage on SyntheticPizza10 → RealPizza10 based on the Faster R-CNN FSOD backbone.

Effectiveness of Progressive Adaptation. In our warm-up stage, each adaptation stage (from FSOD-2 to FSOD-5) provides an improvement of 5.4, 0.6, 4.7, 2.9% compared with the previous step in terms of mAP, respectively. Therefore, each adaptation step in our warm-up stage is helpful.

Impact of Adaptation Order. It is important *when* to use copy-paste augmentation. Starting from the same baseline model FSOD-1, if we sequentially fine-tune the FSOD-1 model on intermediate domain $\mathcal{G}_1$ and augmented intermediate domain $\mathcal{G}_2$, the detection performance will be improved by 6.0% mAP from FSOD-1 to FSOD-3. However, if we sequentially fine-tune the FSOD-1 model on augmented intermediate domain $\mathcal{G}_2$ and intermediate domain $\mathcal{G}_1$, the detection performance will be improved by only 1.6% mAP from FSOD-1 to FSOD-3. Similarly, starting from the same FSOD-3 model, if we sequentially fine-tune the FSOD-3 model on first-round pseudo-labeled domain $\mathcal{T}$ and second-round augmented pseudo-labeled domain $\mathcal{T}$, the detection performance will be improved 7.6% mAP from FSOD-3 to FSOD-5. However, if we sequentially fine-tune the FSOD-3 model on augmented first-round pseudo-labeled domain $\mathcal{T}$ and second-round pseudo-labeled domain $\mathcal{T}$, the detection performance will be improved by 7.0% mAP from FSOD-3 to FSOD-5.

Generalizability of the Warm-up Stage across FSODs. We investigate our warm-up stage on other FSOD models such as Sparse DETR on SyntheticPizza10 → RealPizza10 datasets. Compared with Faster R-CNN backbone, our D^{2}F2WOD$_{\text{warm-up}}$ and D^{2}F2WOD$_{\text{casd}}$ based on Sparse DETR yields 0.6% and 1.1% improvement in terms of mAP, re-

Table 3: Ablation study of D^{2}F2WOD main configurations on Clipart1K → VOC2007 and SyntheticPizza10 → RealPizza10.

		mAP	
Type	Method	Clip → Voc	SPizza → RPizza
SD	OICR	41.2	4.7
D^{2}F2WOD$_{oicr}$	**+FE**	44.7	8.5
	+OP	47.2	12.6
	+FE+OP	52.7	13.8
SD	CASD	57.0	12.9
D^{2}F2WOD$_{casd}$	**+FE**	60.0	14.8
	+OP	60.1	24.0
	+FE+OP	64.8	25.1

spectively. These results emphasize the generality of our framework across different FSOD models. We also conduct ablation studies to investigate the effectiveness of our architecture components in the main FSOD stage, including the domain specific pre-trained deep FE and the weakly-supervised OP generator, as well as the generalization ability of our framework on two WSODs: OICR and CASD. We perform experiments on Clipart1K → VOC2007 and SyntheticPizza10 → RealPizza10. We find that: (1) our domain specific pre-trained deep FE and weakly-supervised OP generator are both necessary for D^{2}F2WOD; and (2) D^{2}F2WOD can generalize to different WSOD methods.

Main Stage Configurations. From Table 3, we observe that compared with the single-domain baseline networks (OICR and CASD), replacing the VGG16 backbone pre-trained on ImageNet with **domain specific pre-trained deep FE** can improve the performance on VOC2007 (mAP from 41.2% to 44.7%, and from 57.0% to 60.0%, respectively), and on RealPizza10 a consistent improvement is achieved with 3.8% and 1.9% for OICR and CASD, respectively. From Table 1, we observe that our object proposal generator is also better than WeakRPN [30], including a two-stage region proposal network. These results confirm the necessity of the domain-specific pre-trained deep features. Table 3 also shows the impact of the **weakly-supervised OP generator**; it achieves consistent improvements of 3.1% and 11.1%, compared with CASD, on VOC2007 and RealPizza10 datasets, respectively. Together, **FE+OP** results in Table 3 suggest that these two key components are both effective and complementary to each other.

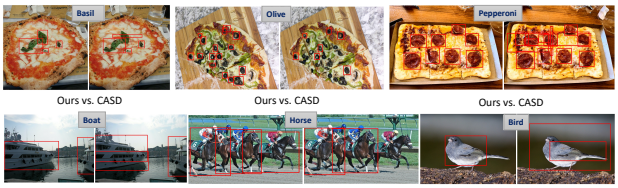

Figure 5: Example of success cases for our D^{2}F2WOD$_{casd}$ vs. CASD in the test set of RealPizza10 and VOC2007 datasets. We only show instances with scores over 0.3 to maintain visibility.

Generalizability of D^{2}F2WOD across WSODs. We investigate the impact of our framework as a function of different WSOD methods (here, OICR and CASD). Results in Table 3 emphasize the **generalizability of D^{2}F2WOD across WSODs**. The performance gain is observed in both OICR and CASD on the two datasets. The effect of D^{2}F2WOD is particularly significant for CASD on RealPizza 10, since our object proposals are target domain-specific and of high-quality. By contrast, existing WSOD methods use hand-crafted selective search to generate object proposals, leading to inaccurate proposals especially for domains such as Pizza that are very dissimilar to VOC2007.

Identifying Object Detection Errors. We use TIDE [3] to understand the classification, localization, both Cls and Loc, duplicate detection, background, and missed GT errors in our model. As shown in Fig. 4, D^{2}F2WOD effectively reduces the localization error. Please see the Sec. 5 of supplementary material for more details.

4.3. Qualitative Analysis

Fig. 5 illustrates the detection results produced by our D^{2}F2WOD and CASD on RealPizza10 and VOC2007 datasets, respectively. There, it can be observed that D^{2}F2WOD does not only locate most objects, but that it also produces more accurate bounding boxes. Specifically, in the RealPizza10 images it can be appreciated bounding boxes provided by our method (left) closely align with the objects of interest, while for CASD (right) bounding boxes are often imprecise (either wrong shape or big/small). Similar observations can be made for VOC2007 where CASD often fails to locate objects or produces spurious bounding boxes.

5. Discussion and Conclusion

We propose D^{2}F2WOD, a simple yet effective object generation strategy that can be applied to different WSOD methods. The key insight is to cast WSOD as a domain adaptation problem and improve performance by progressive foreground-background focused transfer learning of an FSOD from non-photographic source to real-world target domains. Empirical evaluation shows D^{2}F2WOD significantly outperforms state of the art on several benchmarks.

Limitation. Our framework requires extra training time for CycleGAN, which brings in the most additional computation overhead. While D^{2}F2WOD offers a promising way to solve WSOD in the presence of a large domain gap, it currently lacks the ability to jointly learn and refine all stages in the pipeline. An end-to-end large-gap WSOD could offer additional improvement in detection performance on the target domain through creation of increasingly discriminative object features. However, one challenge with that setting would be to control the back-propagation of possible errors induced by the PL steps.

Acknowledgement. This work was supported in part by NSF IIS Grant #1955404.

References

[1] Aditya Arun, CV Jawahar, and M Pawan Kumar. Dissimilarity coefficient based weakly supervised object detection. In *CVPR*, 2019. 2, 6, 7

[2] Hakan Bilen and Andrea Vedaldi. Weakly supervised deep detection networks. In *CVPR*, 2016. 1, 2, 6, 7

[3] Daniel Bolya, Sean Foley, James Hays, and Judy Hoffman. Tide: A general toolbox for identifying object detection errors. In *ECCV*, 2020. 8

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 1, 2, 3, 4

[5] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *CVPR*, 2018. 3

[6] Thomas G Dietterich, Richard H Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial intelligence*, 89(1-2):31–71, 1997. 1

[7] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010. 6

[8] Yan Gao, Boxiao Liu, Nan Guo, Xiaochun Ye, Fang Wan, Haihang You, and Dongrui Fan. C-midn: Coupled multiple instance detection network with segmentation guidance for weakly supervised object detection. In *ICCV*, 2019. 1, 2, 6, 7

[9] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *CVPR*, 2021. 2, 3

[10] Ross Girshick. Fast r-cnn. In *ICCV*, 2015. 2

[11] Han-Kai Hsu, Chun-Han Yao, Yi-Hsuan Tsai, Wei-Chih Hung, Hung-Yu Tseng, Maneesh Singh, and Ming-Hsuan Yang. Progressive domain adaptation for object detection. In *WACV*, 2020. 3, 6, 7

[12] Zitong Huang, Yiping Bao, Bowen Dong, Erjin Zhou, and Wangmeng Zuo. W2n: Switching from weak supervision to noisy supervision for object detection. In *ECCV*, 2022. 2, 6, 7

[13] Zeyi Huang, Yang Zou, BVK Kumar, and Dong Huang. Comprehensive attention self-distillation for weakly-supervised object detection. In *NeurIPS*, 2020. 1, 2, 3, 5, 6, 7

[14] Naoto Inoue, Ryosuke Furuta, Toshihiko Yamasaki, and Kiyoharu Aizawa. Cross-domain weakly-supervised object detection through progressive domain adaptation. In *CVPR*, 2018. 1, 3, 4, 6, 7

[15] Seunghyeon Kim, Jaehoon Choi, Taekyung Kim, and Changick Kim. Self-training and adversarial background regularization for unsupervised domain adaptive one-stage object detection. In *ICCV*, 2019. 3

[16] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICML Workshop*, 2013. 2

[17] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 6

[18] Minh Hoai Nguyen, Lorenzo Torresani, Fernando De La Torre, and Carsten Rother. Weakly supervised discriminative localization and classification: a joint learning process. In *ICCV*, 2009. 2

[19] Dim P Papadopoulos, Youssef Tamaazousti, Ferda Ofli, Ingmar Weber, and Antonio Torralba. How to make a pizza: Learning a compositional layer-based gan model. In *CVPR*, 2019. 1, 6

[20] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 1, 2, 3, 4, 6, 7

[21] Zhongzheng Ren, Zhiding Yu, Xiaodong Yang, Ming-Yu Liu, Yong Jae Lee, Alexander G Schwing, and Jan Kautz. Instance-aware, context-focused, and memory-efficient weakly supervised object detection. In *CVPR*, 2020. 1, 2, 6, 7

[22] Adrian Lopez Rodriguez and Krystian Mikolajczyk. Domain adaptation for object detection via style consistency. *arXiv preprint arXiv:1911.10033*, 2019. 3

[23] Byungseok Roh, JaeWoong Shin, Wuhyun Shin, and Saehoon Kim. Sparse detr: Efficient end-to-end object detection with learnable sparsity. In *ICLR*, 2022. 4, 6

[24] Aruni RoyChowdhury, Prithvijit Chakrabarty, Ashish Singh, SouYoung Jin, Huaizu Jiang, Liangliang Cao, and Erik Learned-Miller. Automatic adaptation of object detectors to new domains using self-training. In *CVPR*, 2019. 3

[25] Kuniaki Saito, Yoshitaka Ushiku, Tatsuya Harada, and Kate Saenko. Strong-weak distribution alignment for adaptive object detection. In *CVPR*, 2019. 3

[26] Vishwanath A Sindagi, Poojan Oza, Rajeev Yasarla, and Vishal M Patel. Prior-based domain adaptive object detection for hazy and rainy conditions. In *ECCV*, 2020. 3

[27] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 3

[28] Peng Tang, Xinggang Wang, Song Bai, Wei Shen, Xiang Bai, Wenyu Liu, and Alan Yuille. Pcl: Proposal cluster learning for weakly supervised object detection. *TPAMI*, 42(1):176–191, 2018. 1, 2, 6, 7

[29] Peng Tang, Xinggang Wang, Xiang Bai, and Wenyu Liu. Multiple instance detection network with online instance classifier refinement. In *CVPR*, 2017. 1, 2, 3, 5, 6, 7

[30] Peng Tang, Xinggang Wang, Angtian Wang, Yongluan Yan, Wenyu Liu, Junzhou Huang, and Alan Yuille. Weakly supervised region proposal network and object detection. In *ECCV*, 2018. 2, 6, 7, 8

[31] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *IJCV*, 104(2):154–171, 2013. 1, 2

[32] Nanne Van Noord and Eric Postma. Learning scale-variant and scale-invariant features for deep image classification. *Pattern Recognition*, 61:583–592, 2017. 2

[33] Vibashan VS, Vikram Gupta, Poojan Oza, Vishwanath A Sindagi, and Vishal M Patel. Mega-cda: Memory guided attention for category-aware unsupervised domain adaptive object detection. In *CVPR*, 2021. 3

[34] Fang Wan, Chang Liu, Wei Ke, Xiangyang Ji, Jianbin Jiao, and Qixiang Ye. C-mil: Continuation multiple instance learning for weakly supervised object detection. In *CVPR*, 2019. 1, 2, 6, 7

[35] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019. 2, 3, 5

[36] Zhaoyang Zeng, Bei Liu, Jianlong Fu, Hongyang Chao, and Lei Zhang. Wsod2: Learning bottom-up and top-down objectness distillation for weakly-supervised object detection. In *ICCV*, 2019. 2, 6, 7

[37] Ming Zhang, Shuaicheng Liu, and Bing Zeng. Hierarchical region proposal refinement network for weakly supervised object detection. In *ICIP*, 2021. 2

[38] Zhi Zhang, Tong He, Hang Zhang, Zhongyue Zhang, Junyuan Xie, and Mu Li. Bag of freebies for training object detection neural networks. *arXiv preprint arXiv:1902.04103*, 2019. 3

[39] Ganlong Zhao, Guanbin Li, Ruijia Xu, and Liang Lin. Collaborative training between region proposal localization and classification for domain adaptive object detection. In *ECCV*, 2020. 3

[40] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 2, 4

[41] C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014. 1, 2